EXHIBIT 2

Load Balancing of Multipath Source Routing in Ad Hoc Networks

Lianfang Zhang, Zenghua Zhao, Yantai Shu, and Lei Wang
Department of Computer Science
Tianjin University, Tianjin 300072, China

Oliver W.W. Yang
School of Information Technology and Engineering
University of Ottawa
Ottawa, Ontario, Canada, K1N 6N5

***Abstract-* A load-balancing scheme has a significant effect on the performance of the multipath routing protocol, especially in an ad hoc network environment. In order to analyze the effect on the distribution of input traffic among multiple paths in MSR (Multipath Source Routing), we first established a network queuing model that would incorporate the cross-traffic among these paths. We then considered the load balancing as an optimization problem. The solution to the optimization problem is interestingly in accordance with the heuristic equation proposed by Wang in ICC'2001 [2]. Our simulation results show that MSR with load balancing is so effective that the end-to-end delay is decreased significantly while the network resource can be utilized more efficiently than that in DSR (Dynamic Source Routing).**

I. Introduction

Mobile ad hoc networks are characterized by multi-hop wireless links, the absence of any cellular infrastructure, and frequent host mobility. Routing protocols in such networks must manage frequent topology changes and need to be bandwidth- and power-efficient. A class of routing protocols called the "on-demand" protocols has recently attracted much attention because of their low routing overhead. An example is the DSR (Dynamic Source Routing)[1]. But this class of protocols is all based on single path routing, which not only under-utilize resources, but also cannot cope with congestion and link breakage. Therefore, using multipath routing can overcome the above problems by providing load balancing and route failure protection by distributing traffic among a set of diverse paths. These benefits make multipath routing a good candidate for mobile bandwidth-limited ad hoc networks.

We have proposed MSR (Multipath Source Routing) algorithm in [2] as an extension of DSR. MSR inherits all the advantages of DSR in addition to the capability of multipath routing. Furthermore, it employs a probing mechanism to obtain on-demand the dynamic path states. This mechanism can be used to refresh the information in cache, to delete stale path and to find new one in time.

How to distribute the traffic over several paths is a key issue in multipath routing [3], and it has a significant effect on the performance of the routing. In [2], a heuristic equation was proposed to balance the traffic load based on an intuitive assumption. The work of [4] analyzed theoretically the characterization of optimal routing, and gave an example of a network with two paths. But their analysis did not consider cross-traffic when solving the load-balancing problem. Unfortunately, bandwidth utilization is very sensitive to cross-traffic in an ad hoc network using MSR. Therefore, in this paper, we have built an analytical model that would consider cross-traffic in order to explain the load-balancing problem in theory, and to provide more details to our original MSR.

The rest of the paper is organized as follows. Section II gives the essence of MSR by summarizing our previous work [2]. Section III gives an analytical model that allows us to obtain some interesting results. The heuristic equation is also discussed in detail. Section IV summarizes the performance evaluation results by simulation. Section V concludes the paper.

II. An Overview To MSR

By using source routing, MSR can improve performance by giving applications the freedom to use multiple paths within the same path service. However, maintaining alternative paths requires more routing table space and computational overhead. Fortunately, some features of DSR can minimize these disadvantages. First, source routing is flexible in that messages can be forwarded on arbitrary paths; this makes it very easy to dispatch messages to multiple paths without demanding path calculation in intermediate hops. Second, the on-demand nature of DSR can reduce greatly the routing storage and routing computation.

A. Path Finding

MSR retains the route discovery mechanism of DSR whereby multiple paths can be returned. Each discovered route is stored in the route cache with a unique route index. So it is easy to select multiple paths from the cache. In multipath routing, path independence is an important property. A more independent path set can offer more aggregate physical resources between a pair of nodes because when those resources are not shared, the less likely the performance of one path would affect the performances of others. To achieve high path independence, disjoint paths are preferred in MSR. There is no looping problem in MSR because the route information is contained inside the packet itself; and any short- or long-lived routing loops can be immediately detected and eliminated.

B. Probing and Load Balancing

In order to monitor real-time information on each path in MSR, we use probing as a feedback control mechanism. We send probing packets periodically to each path, and measure their round-trip time (RTT), and then estimate path delay using Karn algorithm [5]. Note that the delay is an important quantity to reflect the path performance such as congestion. So if a path has a longer delay, less traffic should be dispatched there in order to alleviate congestion. According to the delay of each path, we can distribute traffic over different paths in order to achieve a minimum mean delay for the whole network. This is usually called load balancing. In the next section, we will build an analytical model to discuss this problem.

III. Model Based Analysis

Since MSR uses source routing, intermediate nodes would do nothing except to forward the packet as indicated by the route in

the packet header, thus adding no more processing complexity than DSR. Since all path calculation is done in the source hosts, and optimal routing is intimately related to load balancing, it would be natural to have source nodes to perform load balancing in MSR.

A. *General Analysis on Load Balancing*

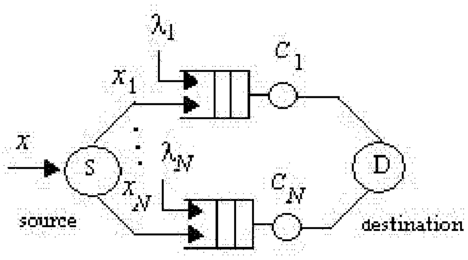

Fig. 1 Network Model

In the network model illustrated in Fig. 1, we show a pair of source and destination nodes with N paths between them. The kth (k=1,2,…,N) path's bottleneck link capacity is assumed to be C_k and the cross-traffic on the kth path is supposed to be Poisson with average arrival rate λ_k. New packet flow is injected into the network from the source node, and split randomly among N paths. Suppose a new packet flow is Poisson with average arrival rate x, then the sub-flow distributed on the kth path is also Poisson [4] with an average arrival rate x_k, such that $x = \sum_{k=1}^{N} x_k$.

Assume the new packet flow and the cross-traffics are independent, and every packet's size has an exponential length distribution, and an average length of $\overline{X}$, then for each constant C_k, each packet service time is also exponential with an average of $\mu_k^{-1} = \dfrac{\overline{X}}{C_k}$.

According to the above assumption, we can model the network as N parallel M/M/1 queues [5]. Our problem now is how to distribute the packet flow x over N paths so as to minimize an object function, i.e. how to achieve load balancing. To do this, we define the objective function as the mean system delay [4]

$$D_T(x_1,\ldots,x_N) = \frac{1}{x + \sum_{k=1}^{N}\lambda_k}\sum_{k=1}^{N} D_k(x_k)$$

where $D_k(x_k) = \dfrac{\lambda_k + x_k}{\mu_k - (\lambda_k + x_k)}$.

Since $D_k(x_k), (k = 1,2,\ldots,N)$ is non-decreasing and convex, we can apply the Lagrangian Multiplier approach to minimize the objective function subject to the constraint $\sum_{k=1}^{N} x_k = x$. Some key steps are provided in the following.

Given the Lagrangian:

$$L(x_1,\ldots,x_N,\gamma) = D_T(x_1,\ldots,x_N) + \gamma\left(x - \sum_{k=1}^{N} x_k\right),$$

where γ is the Lagrangian Multiplier, we differentiate it with respect to x_k, we obtain

$$\frac{d}{dx_k} D_k(x_k) = \gamma \qquad \text{for all} \quad k = 1,2,\ldots,N.$$

It is also easy to show that

$$\frac{d}{dx_i} D_i(x_i) = \frac{d}{dx_j} D_j(x_j), \forall i,j = 1,2,\ldots,N, i \neq j. \qquad (1)$$

Let $x_i^*\ (i = 1,2,\ldots,N)$ be an optimal path flow value, then we can give an explicit form to Eqn. (1) as follows:

$$\frac{1}{\sqrt{u_i}}(\mu_i - \lambda_i - x_i^*) = \frac{1}{\sqrt{u_j}}(\mu_j - \lambda_j - x_j^*), \forall i,j = 1,2,\ldots,N, i \neq j. \qquad (2)$$

The above equations show that in order to achieve optimal load balancing, the available capacity ($\mu_i - \lambda_i$) minus the allocated traffic x_i^* in each path must be proportional to $\sqrt{\mu_i}$. We have thus established another interpretation of load balancing.

To gain more insight into the usage of this new interpretation, let $T_i = \dfrac{1}{\mu_i - \lambda_i}$ be the average packet delay on the ith path before the input traffic x_i is injected. Then we obtain from Eqn. (2)

$$\frac{1}{\sqrt{\mu_i}}\left(x_i - \frac{1}{T_i}\right) = \frac{1}{\sqrt{\mu_j}}\left(x_j - \frac{1}{T_j}\right), \forall i,j = 1,2,\ldots,N, i \neq j. \qquad (3)$$

which is a relationship between the input traffic and the average packet delay on each path. We will demonstrate this with the following example.

B. *Delay Performance Evaluation in Two-path Case*

For the sake of simplicity in illustration, we provide the case of two-path network below.

Following the work developed above, our objective function is

$$D_T(x_1,x_2) = \frac{1}{x + \lambda_1 + \lambda_2}\left[D_1(x_1) + D_2(x_2)\right]$$

where

$$D_1(x_1) = \frac{x_1 + \lambda_1}{\mu_1 - (x_1 + \lambda_1)},$$

$$D_2(x_2) = \frac{x_2 + \lambda_2}{\mu_2 - (x_2 + \lambda_2)}.$$

From (3), we have

$$\frac{1}{\sqrt{\mu_1}}\left(x_1^* - \frac{1}{T_1}\right) = \frac{1}{\sqrt{\mu_2}}\left(x_2^* - \frac{1}{T_2}\right).$$

Under the condition of $x = x_1 + x_2$, and for simplicity, assuming $\mu_1=\mu_2=\mu$, and $T_1<T_2$, it's easy to obtain the solution

$$x_1^* = \frac{1}{2}x + \frac{1}{2}\left(\frac{1}{T_1} - \frac{1}{T_2}\right),$$
$$x_2^* = \frac{1}{2}x + \frac{1}{2}\left(\frac{1}{T_2} - \frac{1}{T_1}\right). \quad (4)$$

The solution is shown in Fig. 2 for x in the range $\left[0, \left(\frac{1}{T_1}+\frac{1}{T_2}\right)\right)$ of possible inputs. There are two regions for the assignment of traffic flow:

1) Region of $0 \le x \le \left(\frac{1}{T_1} - \frac{1}{T_2}\right)$

From (4), we obtain $x_2^* \le 0$, but x_2^* cannot be less than zero, so we must distribute flow such that $x_2^* = 0$, and $x_1^* = x$. It shows that when input traffic is light, no traffic is dispatched to path 2, and only one single path is selected.

2) Region of $\left(\frac{1}{T_1} - \frac{1}{T_2}\right) \le x < \left(\frac{1}{T_1} + \frac{1}{T_2}\right)$

It's easy to deduce that $x_1^* > 0, x_2^* > 0$ in this region. So, when input traffic becomes heavier, both paths are selected, and the input traffic is distributed according to their available capacity. The difference is $x_1^* - x_2^* = \frac{1}{T_1} - \frac{1}{T_2}$

Fig. 2 shows the above results. In summary, we see that when input traffic is light, only the path with shorter delay is used, as the input increases beyond the threshold $\frac{1}{T_1} - \frac{1}{T_2}$, some traffic is routed on the path with a longer delay, and multipath routing is used.

Now, we evaluate the delay performance of the multipath routing. When the input traffic is light $\left(0 \le x \le \left(\frac{1}{T_1} - \frac{1}{T_2}\right)\right)$, multipath routing becomes the single path. While considering optimal routing in $\left(\frac{1}{T_1} - \frac{1}{T_2}\right) \le x < \left(\frac{1}{T_1} + \frac{1}{T_2}\right)$, multipath routing is used. In this case, it can be easily to deduce the system delay D_{T2} :

$$D_{T2} = \frac{1}{x+\lambda_1+\lambda_2}\left[\frac{x_1^*+\lambda_1}{\mu-(x_1^*+\lambda_1)} + \frac{x_2^*+\lambda_2}{\mu-(x_2^*+\lambda_2)}\right]$$
$$= \frac{1}{\mu - \frac{1}{2}(x+\lambda_1+\lambda_2)}. \quad (5)$$

But when we use single path routing, Since $T_1<T_2$, only path 1 is selected, and the average system delay D_{T1} is $D_{T1} = \frac{1}{\mu-(x+\lambda_1)}$.

Since $x \ge \frac{1}{T_1} - \frac{1}{T_2} = \lambda_2 - \lambda_1$, after substituting $\lambda_2 \le x + \lambda_1$ to (5), we obtain

$$D_{T2} \le \frac{1}{\mu - \frac{1}{2}(x+\lambda_1+x+\lambda_1)} = \frac{1}{\mu-(x+\lambda_1)} = D_{T1}$$

i.e. $D_{T2} \le D_{T1}$.

Till now, it's easy to tell that, when the input traffic is larger than a threshold, it's optimal to route the traffic on two paths rather than on a single path.

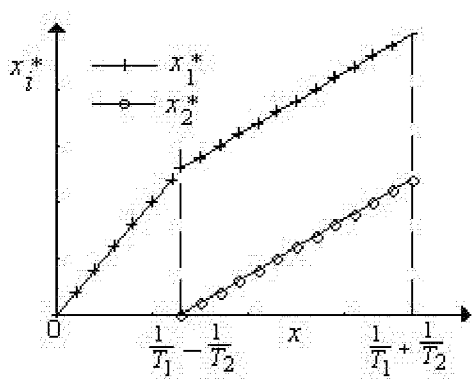

Fig. 2 Optimal path flows of two -path network similar to model in Fig. 1

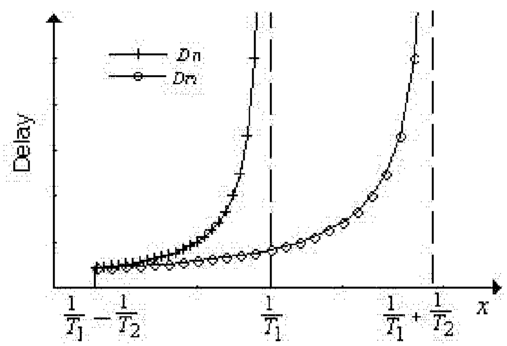

Fig. 3 Comparison of delay between multipath routing and single path one. D_{T1} and D_{T2} are average delay of single path routing and multipath routing respectively

Fig. 3 illustrates intuitively the conclusion that, when the input traffic becomes heavier, multipath routing can decrease the system average delay effectively as well as increase the traffic level, thus utilizing network resources very well.

C. Discussion on the Heuristic Equation

From the above derivation and discussion, we can deduce the method to distribute the traffic over the two paths. When input traffic is light, only one path is chosen. As the traffic increases,

the traffic is dispatched between the two paths. Furthermore, we can conclude that the difference between the two paths $\Delta x = x_1 - x_2 = \frac{1}{T_1} - \frac{1}{T_2} \rightarrow \Delta x \propto \frac{1}{T_1} - \frac{1}{T_2}$.

In general, let $\mu_i = \mu_j = \mu$, Eqn. 3 becomes

$$x_i - x_j \propto \frac{1}{T_i} - \frac{1}{T_j}, \forall i, j = 1,2,\ldots,N, i \neq j. \qquad (6)$$

Then x_i (i=1,2,… N) can be uniquely determined by (6), except for a constant scaling factor (proof is omitted).

Eqn. (6) shows that the difference between the traffic distributed on any two paths must be proportional to the difference of the paths' average packet. In order to make the above distribution method feasible, we propose a load balancing equation.

$$W_k = \frac{1}{d_k} \Big/ \frac{1}{d_{\max}} = \frac{d_{\max}}{d_k}, k = 1,2,\ldots,N$$

where W_k refers to the weight of path k measured in number of packets to be sent consecutively on the same path every time. The variable d_k is the delay of path-k which can be approximated by the RTT [2], and $d_{\max}$ is the maximum delay of all the paths to the same destination,.

The difference between the weights of the two paths is thus

$$\Delta W_{ij} = W_i - W_j = d_{\max}\left(\frac{1}{d_i} - \frac{1}{d_j}\right), \text{ i.e. } \Delta W_{ij} \propto \left(\frac{1}{d_i} - \frac{1}{d_j}\right).$$

much similar to Eqn. (6). Therefore, we may assert that if we distribute traffic according to the weights, we may achieve near-optimal routing. Through a lot of experiment conducted, we have refined our equation to

$$W_k = \min\left(\left\lceil \frac{d_{\max}}{d_k} \right\rceil, U\right) \times R, k = 1,2,\ldots,N. \qquad (7)$$

which is exactly the heuristic equation we proposed in [2]. In (7), U is a bound to insure that W_k should not to be too large. R is a factor to control the frequency of switching between routes. The larger R is, the less frequently the switching would happen, and the less processing overhead required to search and to position an entry in the routing table. When choosing R, the IFQ's (Interface Priority Queue) buffer size should also be taken into considerations. In our experiment, we have found that $R = 3$ to be the best in reducing the out-of-order deliveries in TCP. So R is set to three for an IFQ size of 50. When distributing the load, the weighted-round-robin scheduling strategy is used.

IV. Simulation Result

A. Simulation Environment

We use *ns* [7] to conduct the simulation. CMU has extended *ns* with some wireless supports, including new elements at the physical, link, and routing layers of the simulation environment. Using these elements, it is possible to construct detailed and accurate simulations of the wireless subnets, the LANs, or the multi-hop ad hoc networks. In creating a scenario, two kinds of scenario files are used. The first is a movement pattern file that describes the movement of all mobile nodes in a simulation. The second is a communication pattern file that describes the packet workload offered to the network layer during the simulation.

To get the performance of MSR under different moving speeds, we use two simulation sets with a speed of 1m/s and 20m/s respectively. Our simulations model a network of 50 mobile hosts placed randomly within a 1500m×300m area, both of above two simulation sets are with zero pause time. To evaluate the performance of MSR, we experimented with different application traffics, including CBR and FTP. The CBR application uses UDP as its transport protocol while the FTP uses TCP. The channel is assumed to be error-free except for the presence of collisions. For other simulation detail, please refer to [8].

B. Simulation Results

We first look at the CBR traffic implemented with UDP agents. A scenario with 30 CBR connections is adopted. The maximal moving speed is 20m/s. Since UDP has no feedback control mechanism, all the CBR traffic generated is constant no matter how the network runs. So it can serve as a good reference point for comparing routing protocols. For TCP traffic, we take a scenario with 20 FTP connections and a speed 20m/s.

Fig. 4 and Fig. 5 display the simulation results of CBR and TCP respectively. The figures show the end-to-end delay comparison between the MSR and the DSR algorithms. We can see intuitively that in most connections, the delay of MSR is shorter than that of DSR. Since DSR is single-path-oriented, all the traffic would route along only one path, corresponding to the heavily congested path case. This can be seen from the connection No. 28 in Fig.4, and the connection No. 13 in Fig.5. On the other hand, the traffic on other connections are very light.

Compared with DSR, the advantage of MSR is obvious because it balances the traffic over multiple paths so well that no path is heavily congested. On the contrary, several paths share the traffic with little delay on each path. So through balancing the load among multiple paths, we make good use of network resource and improve end-to-end delay obviously.

V. Conclusion

In this paper, based on our previous work on MSR, we have established a network model in order to study the scheme of load balancing in MSR. We have confirmed our heuristic equation in [2] with a theoretical analysis. By adopting the proposed load-balancing method, the end-to-end delay in MSR has been improved significantly, and the network resource can be utilized efficiently; but our method is still near optimal. The future work might include the study of optimal load balancing scheme and QoS support in MSR.

Acknowledgement

We would like to thank Ping Zhang for her hard work on simulation. Thanks to ns and its CMU wireless extensions. This research was supported in part by the National Natural Science Foundation of China (NSFC) under grant No. 90104015, by a Cisco grant, by an IBM 2001 SUR grant, and by the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant No. OGP0042878.

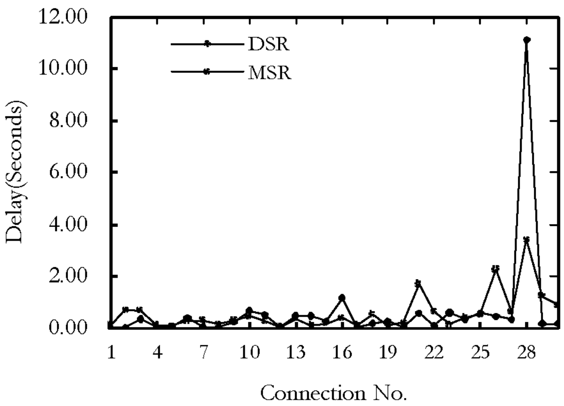

Fig. 4 End-to-end delay of CBR

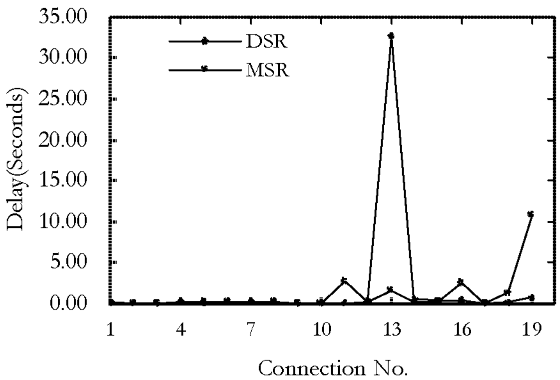

Fig.5 End-to-end delay of TCP

REFERENCES

[1] J. Broch, D. B. Johnson, and D. A. Maltz, "The dynamic source routing protocol for mobile ad hoc networks." http://www.ietf.org/internetdrafts/draft-ietf-manet-dsr-03.txt, June 1999.

[2] L. Wang, L. F. Zhang, Y. T. Shu, M. Dong, and O. W. W. Yang, "Adaptive Multipath Source Routing in Wireless Ad Hoc Networks," *IEEE ICC'98,* Helsinki, Finland, June 2001.

[3] M. Pearlman, Z. Haas, P. Sholander, and S. S. Tabrizi, "On the Impact of Alternate Path Routing for Load Balancing in Mobile Ad-Hoc Networks," *MobiHoc'2000*, Boston, USA, August 11, 2000.

[4] D. Bertsekas, R. Gallager, *Data Networks*, Prentice Hall, 1992

[5] Kleinrock L. *Queueing Systems*, vol. 2. New York: Wiley, 1976

[6] P. Karn and C. Partridge, "Improving Round-Trip Time Estimates in Reliable Transport Protocols", In *ACM SIGCOMM*, pp. 2–7, August 1987.

[7] ns, http://mash.cs.berkeley.edu/ns

[8] J. Broch, D. A. Maltz, D. B. Johnson, Y-C. Hu, and J. Jetcheva. "A performance comparison of multi-hop wireless ad hoc network routing protocols." In *ACM MOBICOM'98*, pp. 85–97, October 1998.